

Défis

Mathématiques avec Scratch

Cette série de défis fait suite aux « Quelques défis pour débiter avec Scratch » (disponibles [ici](#)) qui présentent l'interface et permettent une prise en main progressive. Les défis suivants reprennent la même présentation que les défis de niveau 3 (mais peuvent être réalisés même si vous n'avez pas proposé ces défis à vos élèves) :

- Un mode « libre » : description de ce qui est attendu (de manière assez générale) pour réussir le défi.
L'idée est de laisser les élèves chercher sans les diriger vers l'utilisation de blocs particuliers, l'essentiel étant que leur programme réponde à l'énoncé du défi.
- Un mode « étape par étape » : le défi est décomposé en plusieurs étapes pour guider les élèves qui ne parviendraient pas à réaliser le défi dans le mode précédent. Sous chaque étape, **les éléments de programmation** pouvant être utilisés sont indiqués en italique (ce qui peut aider l'enseignant à choisir les défis en fonctions des éléments déjà rencontrés).
- Une grille d'auto-évaluation associée au mode « étape par étape » (mais qui peut aussi servir d'appui pour les élèves réalisant le défi en mode « libre »).
- Une aide proposant des blocs permettant de réaliser le défi (à ne donner qu'en cas de « blocage » des élèves après un certain temps de recherche et d'essais).
- Une ou plusieurs solutions possibles pour montrer à l'enseignant ce qui peut être attendu.

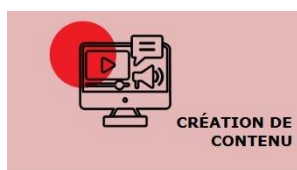
L'idée est de laisser les élèves tâtonner et faire des essais afin de trouver une solution (il peut y en avoir plusieurs) **qui réponde aux missions proposées.**

Ce document est voué à être complété à l'avenir (par moi-même ou par certains d'entre vous ou de vos élèves). N'hésitez pas à me contacter pour me faire parvenir vos remarques ou idées qui, j'en suis persuadé seront très constructives et pertinentes.

Comme l'indique le logo en bas de chaque page, ce document est sous licence *Creative Commons* qui vous autorise à le reproduire, à le partager et à le modifier à conditions que ce ne soit pas à des fins commerciales, que le partage (ou la diffusion) soit effectué sous la même licence que cet original et que l'auteur original soit cité.

Ces défis permettent de travailler les compétences suivantes :

CRCN (Cadre de Référence des Compétences Numériques) :



3.4. Programmer : Écrire des programmes et des algorithmes pour répondre à un besoin (automatiser une tâche répétitive, accomplir des tâches complexes ou chronophages, résoudre un problème logique...) et pour développer un contenu riche (jeu, site web...) (avec des environnements de développement informatique simples, des logiciels de planification de

tâches...).

Compétences transversales :

Domaine 1 : les langages pour penser et communiquer

Comprendre, s'exprimer en utilisant les langages mathématiques, scientifiques et informatiques

Manipuler le clavier pour taper du texte

Domaine 2 : Les méthodes et les outils pour apprendre

Utiliser l'ordinateur

Coopérer et réaliser des projets

Domaine 5 : Les représentations du monde et l'activité humaine

Programmer les déplacements d'un personnage sur un écran

Extrait des programmes (d'après le BOEN n° 31 du 30 juillet 2020)

Au cycle 2, la résolution de problèmes est au centre de l'activité mathématique des élèves, développant leurs capacités à chercher, raisonner et communiquer. Les problèmes permettent d'aborder de nouvelles notions, de consolider des acquisitions, de provoquer des questionnements. Ils peuvent être issus de situations de vie de classe ou de situations rencontrées dans d'autres enseignements, notamment « Questionner le monde », ce qui contribue à renforcer le lien entre les mathématiques et les autres disciplines. Ils ont le plus souvent possible un caractère ludique. On veillera aussi à proposer aux élèves dès le CP des problèmes pour apprendre à chercher qui ne soient pas de simples problèmes d'application à une ou plusieurs opérations mais nécessitent des recherches avec tâtonnements.

Au cycle 3 :

On veille aussi à proposer aux élèves des problèmes pour apprendre à chercher qui ne soient pas directement reliés à la notion en cours d'étude, qui ne comportent pas forcément une seule solution, qui ne se résolvent pas uniquement avec une ou plusieurs opérations mais par un raisonnement et des recherches par tâtonnements.

Sommaire :	Eléments de programmation utilisés dans les défis	p4
	Conseils	p5
	Mission « voiture folle » (NIVEAU 1)	
	<i>Apparence/Déplacement/Boucle/(Repérage)/(Conditions)</i>	
	Mode « libre », mode « étape par étape », grille d'auto-évaluation	p6
	Aide	p7
	Solutions possibles	p8
	Mission « retrouve le chiffre 5 » (NIVEAU 2)	
	<i>Apparence/Communication/Déplacement/Programmation de touches/Condition /Test logique/Temporisation/Boucle</i>	
	Mode « libre », mode « étape par étape »	p9
	Grille d'auto-évaluation, aide	p10
	Solutions possibles	p11
	Mission « table de multiplication par 5 » (NIVEAU 2)	
	<i>Apparence/Boîte de dialogue/Boucle/Test logique/Conditions/Communication</i>	
	Mode « libre », grille d'auto-évaluation	p12
	Mode « étape par étape »	p13
	Aide	p14
	Solutions possibles	p15
	Mission « nombre mystère »	
NIVEAU 2 :	<i>Boîte de dialogue/Condition/Test logique/Communication/Temporisation/Boucle</i>	
	Mode « libre », mode « étape par étape »	p16
	Grille d'auto-évaluation, aide	p17
NIVEAU 3 :	<i>Nombre aléatoire/Variable/Boîte de dialogue/Condition/Test logique/Communication /Temporisation/Boucle</i>	
	Mode « libre », mode « étape par étape »	p18
	Grille d'auto-évaluation, aide	p19
	Solutions possibles (niveaux 2 et 3)	p20
	Mission « trace carré » (NIVEAU 1)	
	<i>Apparence/Repérage/Déplacement/Stylo/(Boucle)</i>	
	Mode « libre », « étape par étape », auto-évaluation	p21
	Aide	p22
	Solutions possibles	p23
	Mission « trace triangle » (NIVEAU 2)	
	<i>Apparence/Repérage/Déplacement/Stylo/(Boucle)</i>	
	Mode « libre », « étape par étape », auto-évaluation	p24
	Aide	p25
	Solutions possibles	p26

Eléments de programmation utilisés dans les défis

Déplacement :

Utilisation de blocs permettant le déplacement et l'orientation d'un sprite.

Repérage :

Utilisation des coordonnées (x ; y) dans le repère orthonormé.

Communication :

Un sprite communique un message écrit (bulle) ou oral (enregistrement d'un son).

Apparence :

Action sur les costumes d'un sprite, sur les arrière-plans ou utilisation des effets graphiques.

Boucle :

Utilisation d'au moins une boucle (répéter ...).

Programmation de touches :

Certaines touches du clavier sont programmées pour déclencher des actions.

Condition :

Utilisation de capteurs permettant de vérifier si une condition est remplie ou pas.

Temporisation :

Gestion des lancements des actions pour qu'elles démarrent à un moment précis ou durent un temps défini.

Boîte de dialogue :

Utilisation d'une boîte de dialogue dans laquelle l'utilisateur peut entrer des données (une « réponse »).

Test logique :

Utilisation des blocs « opérateurs » permettant de comparer deux éléments (texte ou numérique).

Nombre aléatoire :

Utilisation de nombres aléatoires

Variable :

Utilisation d'une variable (valeur que l'on fait varier au cours du programme ; par exemple un score ou un nombre choisi aléatoirement à un moment du programme).

Extension Stylo :

Nécessite l'ajout de l'extension « Stylo » à la liste des catégories de commandes.

NIVEAU		Déplacement	Repérage	Communication	Apparence	Boucle	Programmation de touches	Condition	Temporisation	Boîte de dialogue	Test logique	Nombre aléatoire	Variable	Extension Stylo
1	Voiture folle													
2	Retrouve le 5													
2	Table de 5													
2	Nombre mystère													
3														
1	Trace carré													
2	Trace triangle													

Conseils :

- ❖ **Si vos élèves n'ont jamais utilisé Scratch**, il peut être intéressant (voire nécessaire) de **faire une séance préalable de découverte de l'interface**. Cette séance peut être menée en classe entière si vous disposez d'un TBI (ou simplement d'un vidéoprojecteur). Laissez les élèves observer, puis demandez-leur de décrire ce qu'ils voient. Profitez-en pour leur **apporter le vocabulaire précis** pour faciliter les futurs échanges (il est plus simple de trouver le drapeau vert si on dit qu'il est « en haut à gauche de la scène » plutôt que de dire qu'il est « presque au milieu en haut »...). Vous pouvez vous reporter au document [Quelques défis pour débiter avec Scratch](#).
- ❖ Par expérience, **l'organisation la plus efficace** pour faire réaliser ces défis aux élèves est de les mettre par **groupe de 2** (en veillant à ce qu'il n'y ait pas de trop grande différence de niveau entre ces 2 élèves). Il peut parfois être nécessaire d'instaurer des règles pour établir un « tour de manipulation de l'ordinateur » en expliquant l'importance du rôle de l'élève qui ne manipule pas pour vérifier ce qui est fait et apporter son aide. Par exemple, on peut proposer un changement à chaque étape d'un défi (un élève prend la souris pour l'étape a) et l'autre pour l'étape b).
- ❖ Comme le temps passe toujours vite quand on fait de la programmation, je vous conseille de prendre un petit temps pour **expliquer aux élèves comment enregistrer leur projet avant de commencer à programmer** (ce qui fera gagner du temps en fin de séance).
 - Avec le logiciel installé :
Leur indiquer dans quel dossier enregistrer leur programme et le nom à lui donner. Par exemple, les prénoms des élèves créateurs, suivis du titre du projet. Une fois la séance terminée, ils auront juste à cliquer sur « Sauvegarder sur votre ordinateur », puis sur « Enregistrer » dans la fenêtre qui va s'ouvrir et enfin sur « Oui » pour confirmer le remplacement de la version précédente.
 - Avec la version en ligne : vous avez 2 possibilités pour enregistrer :
 1. **Enregistrer les projets en ligne** : créez un compte pour la classe au préalable, connectez-vous, puis remplacez « Untitled » par les prénoms des élèves créateurs, suivis du titre du projet. Cliquez sur « Fichier » puis « Enregistrer maintenant ». Vous retrouverez les projets enregistrés en ligne en cliquant sur le dossier un haut à droite (à côté de votre identifiant de connexion).
 2. **Enregistrer les projets sur l'ordinateur** (vous pourrez ensuite les rouvrir en cliquant sur « Importer depuis votre ordinateur ») : le projet sera enregistré dans votre dossier de téléchargement sous le nom « Projet Scratch ». Pensez à dire à vos élèves d'aller renommer le fichier en fin de séance (après leur dernier enregistrement) avec leurs prénoms, suivis du titre du projet, pour pouvoir le retrouver plus facilement lors des séances suivantes.
- ❖ **Lors de la première séance**, il vous faudra trouver une organisation de classe (fonctionnement en atelier par exemple) qui vous permettra d'être assez disponible pour être certain que les élèves qui réalisent les défis ne se dispersent pas, pour pouvoir leur donner de petits indices pour les faire avancer et surtout **les inciter à faire des essais**. Comme indiqué dans l'introduction de ce document ne donnez les fiches « Aide » qu'en dernier recours ou si vous ne pouvez pas vous rendre disponible pour aider les élèves « bloqués ». **Pour les séances suivantes, les élèves devraient pouvoir être autonomes**. Vous aurez juste à donner les fiches « Aide » après un délai raisonnable de recherche infructueuse.
- ❖ Si vous en avez la possibilité, imprimez les fiches « Aide » en couleurs (elles seront plus lisibles pour les élèves et leur permettront de trouver plus facilement les blocs à utiliser) et plastifiez-les afin de pouvoir les conserver dans le temps.

Et après ?

Rendez-vous sur le site Numérique éducatif 80 (<http://tice.dsden80.ac-amiens.fr/>) pour retrouver les défis programmation proposés par l'équipe des ERUN de la Somme depuis 2017.

NIVEAU 1

Mission « voiture folle »

Mode « libre » :

Programme les allers-retours de la voiture « Convertible 2 » sur la route de l'arrière-plan « Night City With Street ».



NIVEAU 1

Mission « voiture folle »

Mode « étape par étape » :

a) **Insère l'arrière-plan *Night City With Street* et le sprite *Convertible 2*.
*Apparence***

b) **Quand le drapeau vert est cliqué, la voiture répète indéfiniment :**

- avance jusqu'au bord droit de l'écran
- fait demi-tour quand elle touche le bord de l'écran
- avance jusqu'au bord gauche de l'écran.

Déplacement / Boucle / (Repérage) / (Condition)



NIVEAU 1

Mission « voiture folle »

Auto-évaluation		Avec l'aide	
a) J'ai réussi à insérer l'arrière-plan « Night City With Street ».			
a) J'ai réussi à insérer le sprite « Convertible 2 »..			
b) J'ai réussi programmer le déplacement de la voiture jusqu'au bord droit de l'écran quand le drapeau vert est cliqué.			
b) J'ai réussi à programmer le « demi-tour » de la voiture quand elle touche le bord de l'écran.			
b) J'ai réussi programmer le déplacement de la voiture jusqu'au bord gauche de l'écran.			
b) J'ai réussi à programmer la répétition des allers-retours indéfiniment.			

NIVEAU 1

Mission « voiture folle »

Aide (blocs pouvant être utilisés) :

A collection of Scratch blocks for the 'voiture folle' mission. The blocks are arranged in two columns. The left column contains: 'avancer de [] pas', 'rebondir si le bord est atteint', 'touche le bord [] ?' (with a dropdown menu), 'avancer de [] pas', 'tourner [] de [180] degrés', 'fixer le sens de rotation [gauche-droite]'. The right column contains: 'quand [] est cliqué', 'répéter indéfiniment', 'quand [] est cliqué', 'répéter indéfiniment', 's'orienter en direction de [90]', 'glisser en [1] secondes à x: [] y: []', 'si [] alors', 'répéter indéfiniment', 'si [] alors', 'fixer le sens de rotation [gauche-droite]'. A dashed line with a scissors icon is below the blocks.

NIVEAU 1

Mission « voiture folle »

Aide (blocs pouvant être utilisés) :

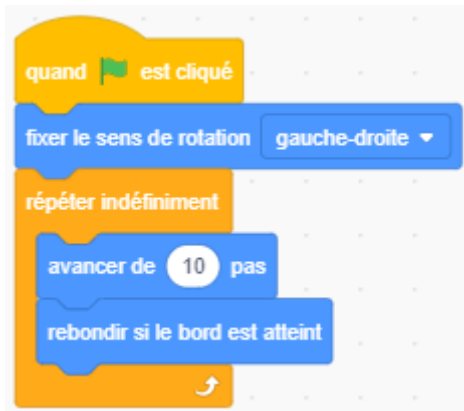
A collection of Scratch blocks for the 'voiture folle' mission. The blocks are arranged in two columns. The left column contains: 'avancer de [] pas', 'rebondir si le bord est atteint', 'touche le bord [] ?' (with a dropdown menu), 'avancer de [] pas', 'tourner [] de [180] degrés', 'fixer le sens de rotation [gauche-droite]'. The right column contains: 'quand [] est cliqué', 'répéter indéfiniment', 'quand [] est cliqué', 'répéter indéfiniment', 's'orienter en direction de [90]', 'glisser en [1] secondes à x: [] y: []', 'si [] alors', 'répéter indéfiniment', 'si [] alors', 'fixer le sens de rotation [gauche-droite]'. A dashed line with a scissors icon is below the blocks.

NIVEAU 1

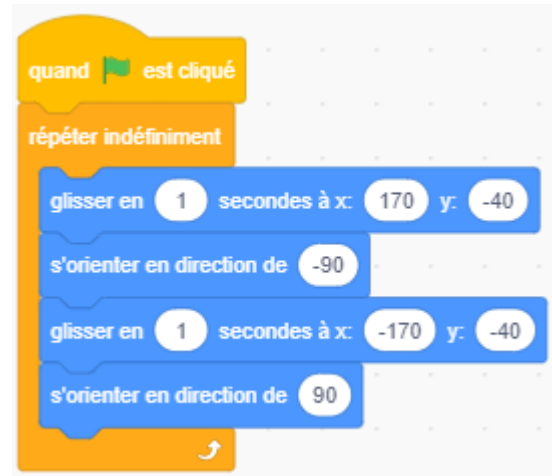
Mission « voiture folle »

Solutions possibles :

Solution 1



Solution 2



Solution 3



NIVEAU 2

Mission « Retrouve le chiffre cinq »

Mode « libre » :

Programme un jeu qui permet de déplacer un personnage : le joueur doit déplacer le sprite « Ladybug2 » jusqu'au chiffre cinq (Sprite « Glow-5 ») à l'aide des flèches du clavier sans toucher les autres chiffres.



NIVEAU 2

Mission « Retrouve le chiffre cinq »

Mode « étape par étape » :

a) **Insère le sprite *Ladybug2* ainsi que les sprites *Glow-0, 1, 2, 3, 4, 5, 6, 7, 8 et 9*. Répartir les sprites dans tout l'écran.**
Apparence

b) **Programme *Ladybug2* pour qu'il dise : « Déplace-moi jusqu'au chiffre cinq »**
Communication

c) **Programme le sprite *Ladybug2* pour qu'il se déplace quand le joueur appuie sur les flèches du clavier.**
Déplacement / Programmation des touches

d) **Le sprite *Ladybug2* dit : « Bravo ! » quand il touche le sprite *Glow-5* et le programme s'arrête.**
Condition / Test logique / Communication / Temporisation / Boucle

e) **Le sprite *Ladybug2* dit : « Essaie encore ! » quand il touche les autres sprites *Glow*.**
Condition / Test logique / Communication / Temporisation / Boucle

NIVEAU 2

Mission « Retrouve le chiffre cinq »

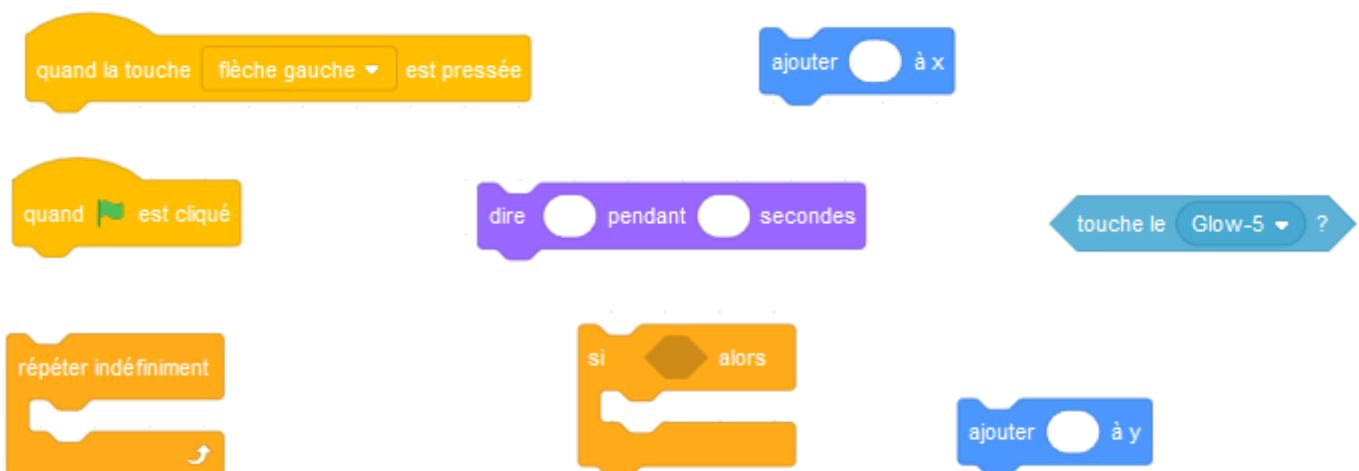
Auto-évaluation	✓	Avec l'aide	✗
a-b) J'ai réussi à programmer l'affichage de la consigne de Ladybug2 quand on clique sur le drapeau vert.			
c) J'ai réussi à programmer le déplacement de Ladybug2 vers le haut, le bas, la droite et la gauche quand on appuie sur les différentes flèches du clavier.			
d) J'ai réussi à programmer l'affichage « Bravo ! » si Ladybug2 touche le sprite Glow-5.			
e) J'ai réussi à programmer l'affichage « Essaie encore ! » si Ladybug2 touche les autres sprites.			



NIVEAU 2

Mission « Retrouve le chiffre cinq »

Aide (blocs pouvant être utilisés) :



NIVEAU 2

Mission « Retrouve le chiffre cinq »

Solutions possibles :

Solution 1

```
quand la touche flèche gauche est pressée
ajouter -10 à x

quand la touche flèche droite est pressée
ajouter 10 à x

quand la touche flèche bas est pressée
ajouter -10 à y

quand la touche flèche haut est pressée
ajouter 10 à y
```

Solution 2

```
quand le drapeau est cliqué
dire Déplace-moi jusqu'au chiffre cinq pendant 2 secondes
répéter indéfiniment
si touche le Glow-5 ? alors
ajouter 10 à x
si touche le Glow-8 ? alors
dire Bravo! pendant 2 secondes
stop tout
si touche le Glow-0 ? alors
dire Essaie encore! pendant 2 secondes
si touche le Glow-5 ? alors
ajouter 10 à x
si touche le Glow-8 ? alors
ajouter -10 à x
si touche le Glow-0 ? alors
ajouter 10 à y
si touche le Glow-5 ? alors
ajouter -10 à y
si touche le Glow-5 ? alors
dire Bravo! pendant 2 secondes
stop tout
si touche le Glow-8 ? alors
dire Essaie encore! pendant 2 secondes
si touche le Glow-0 ? alors
dire Essaie encore! pendant 2 secondes
```

NIVEAU 2

Mission « Table de multiplication par cinq »

Mode « libre » :

Programme un jeu qui demande au joueur les résultats de la table de multiplication par 5.



NIVEAU 2


Mission « Table de multiplication par cinq »

Auto-évaluation		Avec l'aide	
a-b) J'ai réussi à insérer et placer tous les sprites.			
c) J'ai réussi à cacher tous les sprites sauf 0x5 quand on clique sur le drapeau vert.			
d) J'ai réussi à programmer l'affichage de la question du sprite <i>Glow-X</i> quand on clique sur le drapeau vert et l'ouverture d'une boîte de dialogue dans laquelle l'utilisateur peut entrer une réponse.			
e) J'ai réussi à envoyer un message à tous les sprites quand la réponse entrée par le joueur est résultat de 0x5.			
e) J'ai réussi à cacher le 0 quand la réponse est correcte.			
e) J'ai réussi à montrer le 1 quand la réponse est correcte.			
f) J'ai réussi à programmer l'affichage de la question du sprite <i>Glow-X</i> et l'ouverture d'une boîte de dialogue dans laquelle l'utilisateur peut entrer une réponse.			
g) J'ai réussi à envoyer un <u>nouveau</u> message à tous les sprites quand la réponse entrée par le joueur est résultat de 1x5.			
g) J'ai réussi à cacher le 1 quand la réponse est correcte.			
g) J'ai réussi à montrer le 2 quand la réponse est correcte.			
h) J'ai réussi à programmer l'affichage de la question du sprite <i>Glow-X</i> et l'ouverture d'une boîte de dialogue dans laquelle l'utilisateur peut entrer une réponse après chaque réponse correcte pour les opérations 2x5, 3x5, 4x5, 5x5, 6x5, 7x5, 8x5, 9x5.			
h) J'ai réussi à cacher et montrer les bons sprites après chaque réponse correcte.			
i) J'ai réussi à programmer l'affichage d'une bulle « <i>Bravo ! Clique sur le drapeau vert pour recommencer</i> » quand la réponse correcte à 10x5.			

NIVEAU 2

Mission « Table de multiplication par cinq »

Mode « étape par étape » :

- | | |
|----|--|
| a) | <p>Supprime le chat Scratch (sprite1) puis insère les sprites <i>Glow-0, 1, 2, 3, 4, 5, 6, 7, 8</i> et <i>9</i>. Insère encore les sprites <i>Glow-0,1</i> (pour écrire 10) et <i>5</i> ainsi que la lettre <i>Glow-X</i>.</p> <p><i>Apparence</i></p> |
| b) | <p>Place les sprites <i>Glow-0, 1, 2, 3, 4, 5, 6, 7, 8</i> et <i>9</i> les uns au-dessus des autres puis le <i>X</i> et le deuxième <i>5</i> pour qu'ils soient placés comme ceci : </p> <p><i>Apparence</i></p> |
| c) | <p>Programme les sprites à gauche du <i>X</i> pour qu'ils soient cachés quand le drapeau vert est cliqué et que l'on ne voit que <i>0 X 5</i>.</p> <p><i>Apparence</i></p> |
| d) | <p>Programme le sprite <i>Glow-X</i> pour qu'une boîte de dialogue demandant « Quel est le résultat ? » s'ouvre quand le drapeau vert est cliqué (afin que le joueur puisse entrer le résultat de la multiplication) <u>jusqu'à ce que le joueur donne la bonne réponse</u>.</p> <p><i>Aide</i> : Utilise des blocs des catégories <i>Capteurs</i> et <i>Opérateurs</i>.
<i>Boîte de dialogue</i> / <i>Boucle</i> / <i>Condition</i> / <i>Test logique</i></p> |
| e) | <p>Quand la réponse est correcte le sprite <i>Glow-X</i> envoie un message à tous. Quand le sprite <i>0</i> reçoit le message, il se cache et quand le sprite <i>1</i> reçoit le message, il se montre.</p> <p><i>Apparence</i> / (<i>Condition</i> / <i>Test logique</i>)</p> |
| f) | <p>Programme le sprite <i>Glow-X</i> pour qu'une boîte de dialogue demandant « Quel est le résultat ? » s'ouvre (afin que le joueur puisse entrer le résultat de la multiplication 1x5) <u>jusqu'à ce que le joueur donne la bonne réponse</u>.</p> <p><i>Aide</i> : Utilise des blocs des catégories <i>Capteurs</i> et <i>Opérateurs</i>.
<i>Boîte de dialogue</i> / <i>Boucle</i> / <i>Test logique</i></p> |
| g) | <p>Quand la réponse est correcte le sprite <i>Glow-X</i> envoie un <u>nouveau message</u> à tous. Quand le sprite <i>1</i> reçoit le message, il se cache et quand le sprite <i>2</i> reçoit le message, il se montre.</p> <p><i>Apparence</i> / (<i>Condition</i> / <i>Test logique</i>)</p> |
| h) | <p>Répète les étapes f) et g) en les adaptant pour programmer les ouvertures de boîte de dialogue et l'affichage des opérations suivantes à chaque réponse correcte.</p> |
| i) | <p>Quand la réponse de 10x5 est correcte le sprite <i>Glow-X</i> dit « <i>Bravo ! Clique sur le drapeau vert pour recommencer</i> ».</p> <p><i>Communication</i></p> |

NIVEAU 2

Mission « Table de multiplication par cinq »

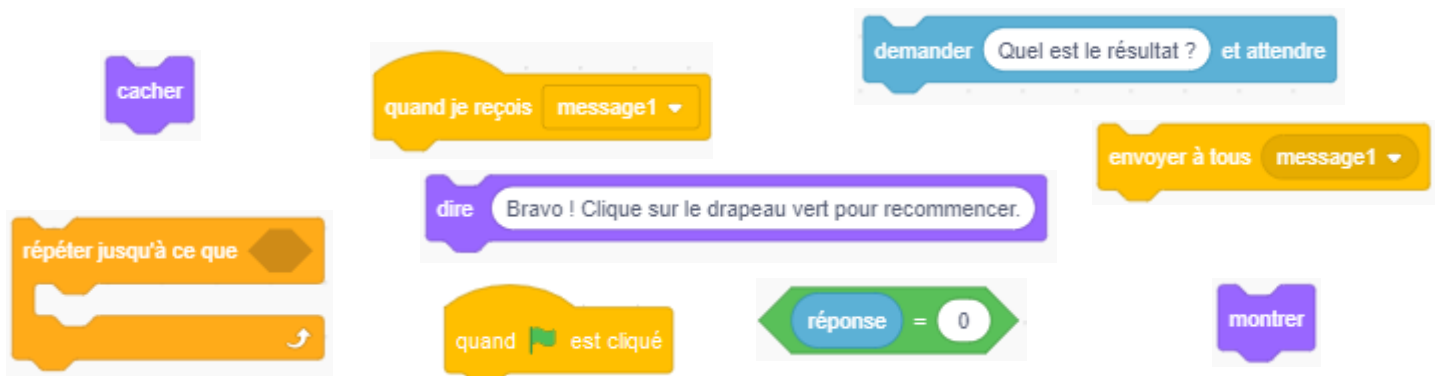
Aide (blocs pouvant être utilisés) :



NIVEAU 2

Mission « Table de multiplication par cinq »

Aide (blocs pouvant être utilisés) :



NIVEAU 2

Mission « Table de multiplication par cinq »

Solution possible :

The code is organized into two columns:

- Left Column (Main Logic):**
 - when green flag is clicked
 - loop: "répéter jusqu'à ce que" (réponse = 0) with "demander" (Quel est le résultat ? et attendre) and "envoyer à tous" (message1)
 - loop: "répéter jusqu'à ce que" (réponse = 5) with "demander" (Quel est le résultat ? et attendre) and "envoyer à tous" (2)
 - loop: "répéter jusqu'à ce que" (réponse = 10) with "demander" (Quel est le résultat ? et attendre) and "envoyer à tous" (3)
 - loop: "répéter jusqu'à ce que" (réponse = 15) with "demander" (Quel est le résultat ? et attendre) and "envoyer à tous" (4)
 - loop: "répéter jusqu'à ce que" (réponse = 20) with "demander" (Quel est le résultat ? et attendre) and "envoyer à tous" (5)
 - (...)
 - loop: "répéter jusqu'à ce que" (réponse = 45) with "demander" (Quel est le résultat ? et attendre) and "envoyer à tous" (10)
 - loop: "répéter jusqu'à ce que" (réponse = 50) with "demander" (Quel est le résultat ? et attendre)
 - dire: "Bravo ! Clique sur le drapeau vert pour recommencer."
- Right Column (Instance Logic):**
 - Instance 0: when green flag is clicked (montrer), when receives message1 (cacher)
 - Instance 1: when green flag is clicked (cacher), when receives message1 (montrer), when receives 2 (cacher)
 - Instance 2: when green flag is clicked (cacher), when receives 2 (montrer), when receives 3 (cacher)
 - (...)
 - Instance 9: when green flag is clicked (cacher), when receives 9 (montrer), when receives 10 (cacher)

NIVEAU 2

Mission « nombre mystère »

Mode « libre » :

Programme un jeu qui permet de trouver un nombre mystère (**choisi par le programmeur**) : le joueur propose un nombre et le programme lui indique si ce nombre est plus grand ou plus petit que le nombre mystère jusqu'à ce qu'il trouve la bonne réponse.



NIVEAU 2

Mission « nombre mystère »

Mode « étape par étape » :

- a) **Scratch demande : « Quel est le nombre mystère ? » quand le drapeau vert est cliqué et une boîte de dialogue s'ouvre pour pouvoir entrer une réponse (utilise un bloc de la catégorie *Capteurs*).**
Boîte de dialogue
- b) **Si la réponse est correcte (c'est-à-dire égale au nombre mystère choisi par le programmeur) Scratch dit « Gagné ! » et le programme s'arrête.**
Condition / Test logique / Communication
- c) **Si la réponse est supérieure au nombre mystère, Scratch dit « Trop grand ! » puis demande à nouveau : « Quel est le nombre mystère ? ».**
Condition / Test logique / Communication / Temporisation / Boucle
- d) **Si la réponse est inférieure au nombre mystère, Scratch dit « Trop petit ! » puis demande à nouveau : « Quel est le nombre mystère ? ».**
Condition / Test logique / Communication / Temporisation / Boucle

NIVEAU 2

Mission « nombre mystère »

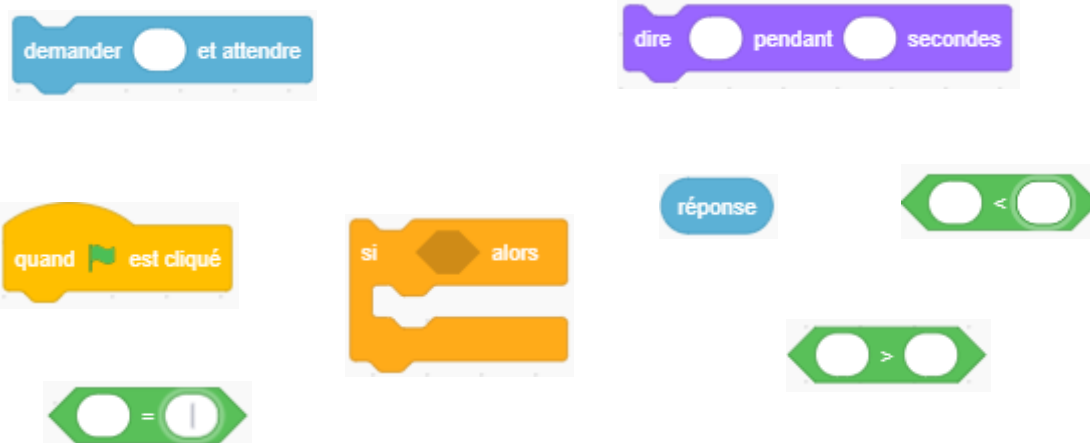
Auto-évaluation	✓	Avec l'aide	✗
a) J'ai réussi à programmer l'affichage de la question de Scratch quand on clique sur le drapeau vert et l'ouverture d'une boîte de dialogue dans laquelle le joueur peut entrer une réponse.			
b) J'ai réussi à afficher la bulle « Gagné ! » quand le nombre mystère est trouvé.			
b) J'ai réussi à arrêter le programme quand le nombre mystère est trouvé.			
c) J'ai réussi à afficher la bulle « Trop grand ! » quand la réponse est supérieure au nombre mystère.			
d) J'ai réussi à afficher la bulle « Trop petit ! » quand la réponse est supérieure au nombre mystère.			
c-d) J'ai réussi à programmer une boucle qui s'arrête quand le nombre mystère est trouvé (Scratch pose la question jusqu'à ce que le nombre mystère soit tapé comme réponse).			



NIVEAU 2

Mission « nombre mystère »

Aide (blocs pouvant être utilisés) :



NIVEAU 3

Mission « nombre mystère »

Mode « libre » :

Programme un jeu qui permet de trouver un nombre mystère (**choisi aléatoirement entre 1 et 30**) : le joueur propose un nombre et le programme lui indique si ce nombre est plus grand ou plus petit que le nombre mystère jusqu'à ce qu'il trouve la bonne réponse.



NIVEAU 3

Mission « nombre mystère »

Mode « étape par étape » :

- a) **Quand le drapeau vert est cliqué un nombre aléatoire entre 1 et 30 et « stocké » (mis en mémoire) dans une variable (ce sera le nombre mystère).**
Nombre aléatoire / Variable
- b) **Scratch demande : « Quel est le nombre mystère ? » et une boîte de dialogue s'ouvre pour pouvoir entrer une réponse (utilise un bloc de la catégorie *Capteurs*).**
Boîte de dialogue
- c) **Si la réponse est correcte (c'est-à-dire égale au nombre aléatoire « stocké » dans la variable) Scratch dit « Gagné ! » et le programme s'arrête.**
Condition / Test logique / Communication
- d) **Si la réponse est supérieure au nombre mystère, Scratch dit « Trop grand ! » puis demande à nouveau : « Quel est le nombre mystère ? ».**
Condition / Test logique / Communication / Temporisation / Boucle
- e) **Si la réponse est inférieure au nombre mystère, Scratch dit « Trop petit ! » puis demande à nouveau : « Quel est le nombre mystère ? ».**
Condition / Test logique / Communication / Temporisation / Boucle

NIVEAU 3

Mission « nombre mystère »

Auto-évaluation	✔	Avec l'aide	✘
a) J'ai réussi à programmer une variable égale à un nombre aléatoire entre 1 et 30 quand le drapeau vert est cliqué.			
b) J'ai réussi à programmer l'affichage de la question de Scratch quand on clique sur le drapeau vert et l'ouverture d'une boîte de dialogue dans laquelle le joueur peut entrer une réponse.			
c) J'ai réussi à afficher la bulle « Gagné ! » quand le nombre mystère est trouvé.			
c) J'ai réussi à arrêter le programme quand le nombre mystère est trouvé.			
d) J'ai réussi à afficher la bulle « Trop grand ! » quand la réponse est supérieure au nombre mystère.			
e) J'ai réussi à afficher la bulle « Trop petit ! » quand la réponse est supérieure au nombre mystère.			
d-e) J'ai réussi à programmer une boucle qui s'arrête quand le nombre mystère est trouvé (Scratch pose la question jusqu'à ce que le nombre mystère soit tapé comme réponse).			



NIVEAU 3

Mission « nombre mystère »

Aide (blocs pouvant être utilisés) :



NIVEAU 2

Mission « nombre mystère »

Solution possible :



NIVEAU 3

Mission « nombre mystère »

Solution possible :



NIVEAU 1

Mission « Trace carré »

Mode « libre » :

Programme le tracé d'un carré dont les côtés mesurent 150 pixels (ou « pas »).



NIVEAU 1

Mission « Trace carré »

Mode « étape par étape » :

a) **Insère le sprite *Arrow1* et place-le au milieu de l'écran (à la place de Scratch)**
Apparence / Repérage

b) **Ajoute l'extension « Stylo » pour pouvoir tracer avec le sprite *Arrow1***

Ajouter une extension



c) **Mets le stylo en position d'écriture, oriente le sprite *Arrow1* vers la droite et programme-le pour tracer un segment de 150 pixels de longueur.**
Déplacement / Stylo

d) **Orienté à nouveau le sprite *Arrow1* et programme son déplacement pour tracer le 2ème côté puis le 3ème côté et enfin le quatrième côté de ton carré.**
Déplacement



NIVEAU 1

Mission « Trace carré »

Auto-évaluation		Avec l'aide	
a-b) J'ai réussi à préparer le programme en insérant le sprite <i>Arrow1</i> et en ajoutant l'extension <i>Stylo</i> .			
c) J'ai réussi à programmer l'orientation de <i>Arrow1</i> vers la droite et à tracer un segment de 150 pixels de longueur.			
d) J'ai réussi à orienter <i>Arrow1</i> et à tracer les 3 autres côtés du carré.			

NIVEAU 1

Mission « Trace carré »

Aide (blocs pouvant être utilisés) :



NIVEAU 1

Mission « Trace carré »

Aide (blocs pouvant être utilisés) :



NIVEAU 1

Mission « Trace carré »

Solutions possibles :

Solution 1

```
quand est cliqué
effacer tout
relever le stylo
aller à x: 0 y: 0
stylo en position d'écriture
s'orienter à 90
avancer de 150 pas
tourner de 90 degrés
avancer de 150 pas
tourner de 90 degrés
avancer de 150 pas
tourner de 90 degrés
avancer de 150 pas
```

Avec une temporisation pour mieux voir les déplacements :

```
quand est cliqué
effacer tout
relever le stylo
aller à x: 0 y: 0
stylo en position d'écriture
s'orienter à 90
attendre 1 secondes
avancer de 150 pas
répéter 3 fois
attendre 1 secondes
tourner de 90 degrés
attendre 1 secondes
avancer de 150 pas
```

Solution 2

```
quand est cliqué
effacer tout
relever le stylo
aller à x: 0 y: 0
stylo en position d'écriture
s'orienter à 90
avancer de 150 pas
répéter 3 fois
tourner de 90 degrés
avancer de 150 pas
```

NIVEAU 2

Mission « Trace triangle »

Mode « libre » :

Programme le tracé d'un triangle équilatéral dont les côtés mesurent 150 pixels (ou « pas »).



NIVEAU 2

Mission « Trace triangle »

Mode « étape par étape » :

- a) **Insère le sprite *Arrow1* et place-le au milieu de l'écran (à la place de Scratch)**
Apparence / Repérage
- b) **Ajoute l'extension « Stylo » pour pouvoir tracer avec le sprite *Arrow1***
Ajouter une extension
- c) **Mets le stylo en position d'écriture, oriente le sprite *Arrow1* vers la droite et programme-le pour tracer un segment de 150 pixels de longueur.**
Déplacement / Stylo
- d) **Orienté à nouveau le sprite *Arrow1* et programme son déplacement pour tracer le 2ème côté puis le 3ème côté du triangle équilatéral.**
Déplacement



NIVEAU 2

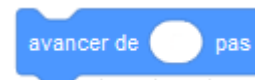
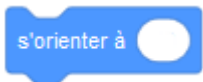
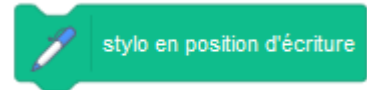
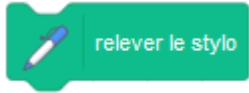
Mission « Trace triangle »

Auto-évaluation		Avec l'aide	
a-b) J'ai réussi à préparer le programme en insérant le sprite <i>Arrow1</i> et en ajoutant l'extension <i>Stylo</i> .			
c) J'ai réussi à programmer l'orientation de <i>Arrow1</i> vers la droite et à tracer un segment de 150 pixels de longueur.			
d) J'ai réussi à orienter <i>Arrow1</i> et à tracer les 2 autres côtés du triangle équilatéral.			

NIVEAU 2

Mission « Trace triangle »

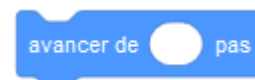
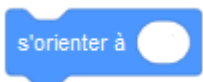
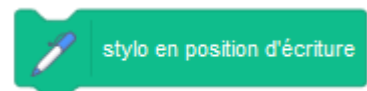
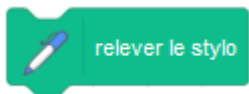
Aide (blocs pouvant être utilisés) :



NIVEAU 2

Mission « Trace triangle »

Aide (blocs pouvant être utilisés) :

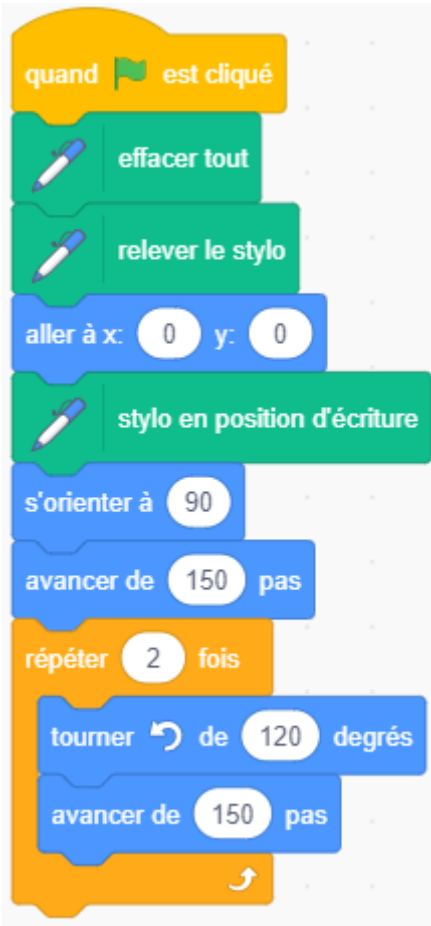


NIVEAU 2

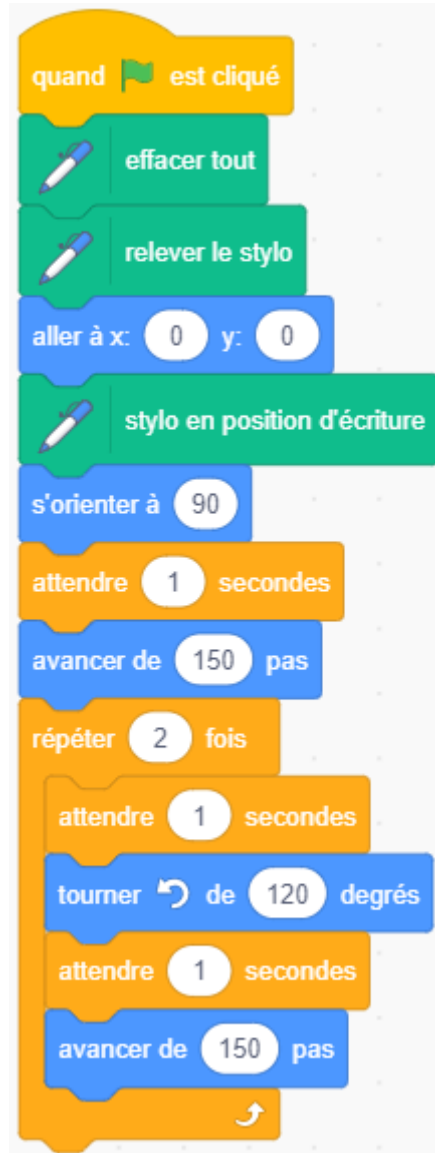
Mission « Trace triangle »

Solutions possibles :

Solution 1



Avec une temporisation pour mieux voir les déplacements :



ou solution 2

